

In the Claims:

Please amend the claims as follows:

1. (currently amended) A method to revalidate a compiler (22) intended for compilation of a user-written program for safety control in an industrial control system (2) after it has been used, ~~comprising the steps of~~ the method comprising:

- compiling (11a) a test program (20) a first time which test program is defined in a control language;
 - validating (11b) the compiler by verifying that the test program executes correctly;
- ~~characterized by the further steps of~~
- generating (12) a first software means derived from the compiled test program intended for later comparison purposes;
 - compiling (13) the test program a second time after the compilation of a user-written program;
 - generating (14) a second software means intended for a comparison based on the second compilation of the test program;
 - comparing (15) the first software means with the second software means; wherein the compiler (22) is revalidated for any errors introduced between the first and the second compilation;
 - enabling (16), provided that the revalidation indicates no errors in the compiler (22), the user-written program to execute in a device (6a) with safety features for control of real world entities (10).

2. (currently amended) ~~A~~ The method according to claim 1, ~~characterized in that~~

wherein the comparing step (15) is performed in the same workstation (5a) or general-purpose computer as that in which the compiler (22) is executing.

3. (currently amended) A The method according to claim 1, ~~characterized in that~~
wherein the software means is a check-sum or a code for cyclic redundancy check.

4. (currently amended) A The method according to claim 3, ~~characterized in that~~
wherein the comparing step (15) is performed in the device (6a) with safety features.

5. (currently amended) A The method according to claim 4, ~~characterized in that~~
wherein the comparing step (15) comprises an additional step of downloading a variable that changes over time, which is downloaded in the same message as the check-sum or code to the device (6a), where the variable that changes over time is used to achieve a change in the message.

6. (currently amended) A The method according to claim 1, ~~characterized in that~~
wherein the test program (20) is defined in a control language derived from the standard IEC 6-1131.

7. (currently amended) A computer program product (5b) containing software code means loadable into the internal memory of a general-purpose computer or workstation (5a) and/or a device (6a), ~~characterized in that~~ wherein said computer program product has means to execute a computer-implemented step of compiling (13) the test program a second time, a

computer-implemented step of generating a second software means (14), a computer-implemented step of comparing (15) the first software means with the second software means and a computer-implemented step of enabling (16) the user-written program to execute in the device (6a), all steps according to claim 1.

8. (currently amended) A The computer program product (~~5b~~) according to claim 7, which comprises software means for carrying out a further action to:
-receive a signal sent across the Internet (~~1~~) comprising the first software means (~~35~~).

9. (currently amended) A computer program comprising computer code means and/or software code portions for making a computer or processor perform any of the steps of claim 1 ~~claims 1-6~~.